AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions of claims in the application:

Listing of Claims:

1. (Currently amended) A system that facilitates remoting services in a distributed object system, the system comprising:

a remote object monitor~~, operable to~~ that monitors a remote object;

a remote object manipulator~~, operably~~ connected to the remote object monitor, the remote object manipulator ~~operable to~~ manipulates the remote object; and

a lifetime manager ~~operable to~~ that controls the lifetime of the remote object, the lifetime manager ~~operably~~ connected to the remote object manipulator and employs a lease to determine a lifetime of the remote object, the lease comprising an initial lease period.

2. (Currently amended) The system of claim 1, where the remote object monitor ~~is~~ further ~~operable to~~ provides a human readable reference to [[a]]the remote object.

3. (Currently amended) The system of claim 2, where the human readable reference to [[a]]the remote object codes information comprising at least one of protocol information, protocol data, an application name and an object URI (Uniform Resource Identifier).

4. (Currently amended) The system of claim 3, where the human readable reference to [[a]]the remote object is a URL (Uniform Resource Locator).

5. (Original) The system of claim 3, where the protocol information is at least one of HTTP (Hypertext Transfer Protocol) information and SMTP (Simple Mail Transfer Protocol) information.

6. (Currently amended) The system of claim 1, where the remote object monitor ~~is operable to~~ acquires metadata concerning [[a]]the remote object.

7. (Currently amended) The system of claim 6, where the metadata comprises at least one of information concerning interfaces implemented by the remote object, a type of the remote object, a class hierarchy of the remote object, methods implemented by [[a]]the remote object, properties implemented by [[a]]the remote object, fields implemented by [[a]]the remote object and attributes implemented by [[a]]the remote object.

8. (Currently amended) The system of claim 1, where the remote object monitor ~~is operable to~~ provides entry points and process interception to facilitate activating a custom attribute based process.

9. (Original) The system of claim 8, where the custom attribute based activated process is performed before non-attribute code associated with a proxy object.

10. (Original) The system of claim 8, where the custom attribute based activated process is performed in parallel with non-attribute code associated with a proxy object.

11. (Original) The system of claim 8, where the custom attribute based activated process is performed after non-attribute code associated with a proxy object.

12. (Original) The system of claim 8, where the custom attribute based activation process is performed at least one of before, in parallel with, and/or after non-attribute code associated with a proxy object.

13. (Previously presented) The system of claim 1, where the remote object monitor and the lifetime manager are implemented within a single component.

14. (Currently amended) The system of claim 13, where the ~~lifetime manager employs a~~ lease ~~manager to monitor the lifetime of a remote object~~ further comprises a renew on access time.

15. (Currently amended) The system of claim 1, where the remote object monitor ~~is operable to~~ provides a human readable reference to [[a]]the remote object, ~~to~~ provides metadata concerning [[a]]the remote object, and ~~to~~ provides entry points and process interception to facilitate activating a custom attribute based process.

16. (Currently amended) The system of claim 1, where the remote object manipulator ~~is operable to~~ updates metadata associated with [[a]]the remote object.

17. (Original) The system of claim 16, where the metadata comprises at least one of information concerning interfaces implemented by the remote object, the type of the remote object, the class hierarchy of the remote object, methods implemented by the remote object, properties implemented by the remote object and attributes implemented by the remote object.

18. (Previously presented) The system of claim 1, where the remote object manipulator and the lifetime manager are implemented within the same component.

19. (Currently amended) The system of claim 18, where the ~~lifetime manager employs a~~ lease ~~manager to control the lifetime of the remote object~~ further comprises a renew on access time.

20. (Currently amended) The system of claim 1, where the remote object manipulator ~~is operable to~~ updates metadata concerning [[a]]the remote object.

21. (Currently amended) A computer readable medium storing computer executable components of a system that facilitates remoting services in a distributed object system, the system comprising:

a remote object monitoring component;

a remote object manipulating component ~~operably~~ connected to the object monitoring component; and

a lifetime managing component ~~operable to~~ controlling the lifetime of a remote object, the lifetime managing component ~~operably~~ connected to the remote object manipulating component and specifying a pre-determined lifetime for the remote object.

22. (Currently amended) A system that provides remoting services in a distributed object system, the system comprising:

an object reference generator ~~operable to~~ produc[[e]]ing a human readable object reference to a remote object;

an object reference extender ~~operable to~~ extending an object reference class subclassed from a base class object reference class;

an interceptor ~~operable to~~ facilitat[[e]]ing activating attribute based processing; and

a lifetime monitor ~~operable to~~ manag[[e]]ing ~~the~~ a lifetime of the remote object *via* a lease having at least an initial lease time.

23. (Currently amended) The system of claim 22, where the human readable object reference to [[a]]the remote object codes information comprising at least one of protocol information, protocol data, an application name and an object URI (Uniform Resource Identifier).

24. (Original) The system of claim 23, where the protocol information is at least one of HTTP (Hypertext Transfer Protocol) information and SMTP (Simple Mail Transfer Protocol) information.

25. (Currently amended) The system of claim 22, where the object reference extender ~~is~~ further ~~operable to~~ facilitates overriding at least one of a method, an attribute, a property, a field, an interface and an event associated with the base class object reference class in the subclassed object reference class.

26. (Currently amended) The system of claim 25, where the object reference extender ~~is~~ further ~~operable to~~ adds at least one of a method, an attribute, a property, a field, an interface and an event to the subclassed object reference class.

27. (Original) The system of claim 22, where the attribute activated based processing is performed at least one of before, substantially in parallel with, and/or after non-attribute code associated with a proxy object.

28. (Currently amended) The system of claim 22, where the ~~lifetime monitor employs a~~ lease ~~manager to monitor the lifetime of the remote object~~ further has a renew on access time.

29. (Currently amended) The system of claim [[28]]22, where ~~the lease manager is further operable to control the lifetime of~~ the remote object specifies the initial lease time.

30. (Currently amended) The system of claim 29, where the ~~lease manager~~ lifetime monitor interacts with a garbage collector to control the lifetime of the remote object.

31. (Currently amended) A computer readable medium storing computer executable components of a system that provides remoting services in a distributed object system, the system comprising:

an object reference generating component ~~operable to~~ produc[[e]]ing a human readable object reference to a remote object;

an object reference extending component ~~operable to~~ extending an object reference class subclassed from a base class object reference class;

an intercepting component ~~operable to~~ facilitate[[e]]ing activating attribute based processing; and

a lifetime monitoring component ~~operable to~~ manag[[e]]ing the lifetime of the remoted object, wherein the lifetime monitoring component sets an initial lease period to control a lifetime of the remote object.

32. (Currently amended) A method for providing remoting services in a distributed object system, the method comprising:

providing an object reference base class from which a derived object reference class can inherit;

providing a human readable object reference to an instance of the object reference base class; and

controlling a lifetime of a remote object by establishing a lease for the remote object comprising an initial lease period.

33. (Original) The method of claim 32, where the object reference base class comprises:

one or more attributes that store information associated with at least one of:

the object type of an instance of the object reference base class;

an envoy associated with the instance of the object reference base class;

a channel associated with the instance of the object reference base class; and

a URI associated with the instance of the object reference base class.

34. (Original) The method of claim 33, where the object reference base class implements one or more interfaces that facilitate at least one of reading, writing and overriding the one or more attributes.

35. (Original) The method of claim 32, further comprising:

inheriting from the object reference base class;

overriding elements of the object reference base class in the derived object reference class; and

adding elements to the derived object reference class.

36. (Original) The method of claim 35 where the elements comprise at least one of a property, a method, an interface, a field, an attribute and an event.

37. (Canceled)

38. (Currently amended) The method of claim 32, where controlling the lifetime of the remote object further comprises:

~~establishing a lease for the remote object;~~

selectively renewing the lease when the remote object is accessed; and

selectively garbage collecting remote objects whose leases have expired.

39. (Original) The method of claim 38, where controlling the lifetime of the remote object further comprises querying a lease sponsor before garbage collecting a remote object whose lease has expired.

40. (Currently amended) The method of claim 32, further comprising:

intercepting calls made on [[a]]the remote object;

determining whether ~~thea~~ proxy has attributes that desire attribute based activation; and

selectively performing attribute based code associated with the proxy.

41. (Currently amended) A method for providing remoting services in a distributed object system, the method comprising:

providing an object reference base class from which a derived object reference class can inherit;

providing a human readable object reference to an instance of the object reference base class;

creating a derived object reference class by inheriting from the object reference base class;

overriding elements of the object reference base class in the derived object reference class;

adding elements to the derived object reference class;

controlling the lifetime of ~~thea~~ remote object *via* a lease comprising an initial lease period;

intercepting calls made on the remote object;

determining whether ~~thea~~ proxy has attributes that desire attribute based activation; and

selectively performing attribute based code associated with the proxy.

42. (Currently amended) A computer readable medium storing computer executable instructions operable to perform a method for providing remoting services in a distributed object system, the method comprising:

providing an object reference base class from which a derived object reference class can inherit;

providing a human readable object reference to an instance of the object reference base class;

creating a derived object reference class by inheriting from the object reference base class;

overriding elements of the object reference base class in the derived object reference class;

adding elements to the derived object reference class;

controlling the lifetime of ~~the~~a remote object *via* a lease comprising an initial lease period;

intercepting calls made on the remote object;

determining whether ~~the~~a proxy has attributes that desire attribute based activation; and

selectively performing attribute based code associated with the proxy.

43. (Currently amended) A system for providing remoting services in a distributed object system, the system comprising:

means for defining a subclassable object reference class;

means for acquiring an instance of the subclassable object reference class;

means for acquiring a human readable reference to the instance;

means for producing a derived object reference class that inherits from the subclassable object reference class;

means for customizing the derived object reference class;

means for creating a lease having an initial lease time that determines the lifetime of a remote object;

means for initiating garbage collection of [[a]]the remote object upon expiration of the lease;

means for intercepting remote method calls; and

means for selectively activating attribute code associated with ~~the~~a proxy.

44. (Currently amended) A data packet adapted to be transmitted between two or more components, the data packet comprising:

a first field that stores information associated with a human readable reference to a remote object; and

a second field that stores information associated with monitoring and controlling ~~the~~a lifetime of the remote object, wherein the information associated with monitoring and controlling the lifetime of the remote object comprises an initial lease period.

45. (Canceled)

46. (Currently amended) The data packet of claim [[45]]44, further comprising:

a third field that stores information associated with attribute activated processing.

47. (Currently amended) A data packet adapted to be transmitted between two or more components, the data packet comprising:

a first field that stores information associated with a human readable reference to a remote object;

a second field that stores information associated with monitoring and controlling ~~the~~a lifetime of the remote object, wherein the information associated with monitoring and controlling the lifetime of the remote object comprises an initial lease period and a renew on access lease time;

a third field that stores information associated with attribute activated processing; and

a fourth field that stores metadata associated with the remote object.